

administrative routine 18 feb 1970

onn=335  
lli=336  
sd=302  
uc=337  
sr0=343  
ntb=344  
ubs=345  
frp=346  
qqt=347  
bop=350  
ctb=351  
srr=356  
mtt=357  
bc=360  
nuc=364  
rs1=365  
cpp=366 /process chain  
c7=40000 /core 7

lok=770040  
ulk=770041

nuf=20 /number of user fields

/process words

dil=6 /dia word  
prn=7 /process ring  
prq=11 /process queue  
  
cms=6630 /origin of computation blocks  
ffs=105 /file capability offset

dimension cns(1),cmp(1),prc(1)  
dimension cpd(1),pdl(40)  
pde=pdl+40

/console words

aw1=0 /assignment word  
msk=3 /console mask  
id=4 /pointer to ID  
rr0=5 /reader switch  
pp0=6 /punch switch  
df1=7 /drum field table

/computation words

quu=11 /computation queue (two words)  
bp1=13 /location of breakpoint  
bp2=bp1+1 /proceed count  
bp3=bp2+1 /instruction under breakpoint  
itr=bp3+1 /illegal instruction return  
imr=itr+1 /illegal memory reference return  
sup=imr+1 /superior sphere  
spe=sup+1 /fault entry to superior  
be1=spe+1 /break enable  
con=be1+2 /pointer back to console  
prh=con+2 /process hoard

define die

jmp .

terminate

```
offset 3200
100/
adr,
jmp .
jmp atm /meta instruction
jmp .
103,    770111 /wait for login/logout job
iam
eem
dac new
lgl,    lac new
xor old
CAIM|
jmp 103
SII
A->IAI
dac t66
xor old
dac old
A->IX
law 37
rir 7s
ril 1s
SAA
spi i
jmp .-3
dap vv1
add (c7 ctb-40
X->AX
lxr i 0
X->AX
ior (c7
dac cns
TXP
jmp on /login

lac t66 /logout
cma
aam
and (c7 onn
aam
sad (c7 onn
jmp lgl /not logged in
aam
dac (c7 onn      /disable call button
```

/be sure to flush core 0 C-list so core 7 stuff can be deleted

```
lac vv1
mta 204 /delete ID and user
lac t66
cma
dac t1
jsp exm
lxr cns
lio i aw1
iot 677 /lar
iot 1577      /scn
rcl 4s
add (c7
```

aam  
add (c7 bop  
TAAX  
llo i 6 /bip  
dio i 0 /bop  
iot 277 /bff  
iot 1177 /ben  
iot 77 /psf  
cli  
iot 677 /lar  
iot 1577 /scn  
jsp lxm  
lxr cns  
aam  
lac (c7 lli  
dac i rr0  
dac i pp0  
lac t1  
and rw  
dac rw  
lac t1  
and pw  
dac pw  
lac t1  
and xrw  
dac xrw /ext reg  
jsp daf  
jsp dxe  
law i 200  
and i aw1  
dac i aw1 /dismiss ext reg  
jmp lg1

on,

```
lac vv1
mta 302 /create ID
jmp noc
lio (030000
law 22
xct vv1 /attach it
die
cla\swp
mta 1
lio (250000
mta 104 /read a copy of ID
jmp .-1
law 101
mta
lac (201272
xct vv1 /set ill. inst. ret. for ID
lac (203272
xct vv1 /set ill. mem. ref. ret.
law 202 /create a process for ID
xct vv1
jmp noc
law idp /initialize process state
mta
law 52
lio (1
xct vv1
die
law 77
and vv1
ior (i
TAX
lac i 0
and (7777
lxr cns
dac i id
ior (c7
X->AX
dzm i sup           /erase its superior
dzm i prh           /set its quota to one
dap i con            /give ID a C-list
lac (403372         /turn on PRL for ID
xct vv1
lio (100
law 14
mta
lac (302272
xct vv1 /create user for ID
jmp noc
lio (1450           /bring user into self at index 50
law 242
xct vv1
die
aam
lac (010050
and (7777
ior (c7
TAX
lac cns
dap i con           /give user a core 7 C-list
```

```
law 202
inv 50 /create process for user
jmp noc
law 50
mta 204 /erase copy of user
law 12
xct vvi /start ID running
lac t66
aam
ior (c7 onn
aam
dac (c7 onn
jmp lgl

noc,
lac vvi /can't log in
mta 204 /delete ID if created it
law 50
mta 204 /delete extra copy of user
jmp lgl
```

/initial process state for ID

idp,	0	102	0
	0	0	0
t66,	0		
new,	0		
old,	0		
vvi,	inv .	/console number+36	

/process list (used only for reference counts)

dimension pof(100)  
dimension lif(100) /hardware device table

/programmed queue list

frq,           qqf+1  
qqf,           1  
             repeat 3,,+1  
             0

/file list

frf,           fff     /free files  
fff,           repeat 37,,+1  
             0

/sphere list

frc,           ccf+2  
ccf,           1       /adm rt  
             1       /tape controller  
             repeat 14,,+1  
             0

/microtape list

mtu,           repeat 4,0

t0,           0  
t,           0  
t1,           0  
t2,           0  
t3,           0  
t4,           0  
t5,           0  
t6,           0  
t7,           0  
t8,           0  
t8+1,        0  
cmq,          0  
cmr,          0

/table to rotate translation word

r1,           ril 3s   ril 6s     ril 9s  
             rir 6s   rir 3s  
r2,          nop

```
/set memory bound for computation in cmr(0)
/t5 = new bound

smb,          0
    lac cmr
    ior (c7
    TAX
    dzm t3
sm1,          law i 7777           /find present bound
    and i 1
    sza i
    jmp .+5
    SXX
    idx t3
    sas (6
    jmp sm1
    lac t3 /old bound
    sub t5 /new bound
    spa
    jmp ccc

    jsp exm /lower the bound
sm2,          lac t5
    add cmr
    dac t
    ior (c7
    dac t1
    sub t5
    dac t2
    lxr t1
    law i 7777
    and i 1
    spa
    jmp dpc /in core
    ral 6s
    and (37
    add (c7 sd-1
    jmp ct3
dpc,          aam
    lio t2
    lxr t5
    xct i r1
    law 7
    A←IA
    add (c7 bc
    TAAAX
    lio (1000
    dio i 0 /give it high swap priority
    add (uc-bc

ct3,          TAX
    dzm i 0
    jdp ct1 /flush this core and attachments
    idx t5
    sas t3
    jmp sm2
    jsp lxm
    idx smb
    jmp i smb
sms,
```

ccc,  
sza i  
jmp sms /bound is unchanged  
  
dac t2 /raise the bound  
jsr exm  
lrx (c7 /count cores  
lio i uc  
sni  
idx t2  
lio i uc+1  
sni  
idx t2  
lac t2  
sma  
jmp sm4 /already have enough  
lrx (c7 sd /count drum fields  
lac i 0  
and (377777  
sza i  
jmp .+5  
SXXA  
sas (c7 sd+26  
jmp sm3  
jmp smy /can't get enough  
isp t2  
jmp .-5  
  
sm3,  
sza i  
jmp .+5  
SXXA  
sas (c7 sd+26  
jmp sm3  
jmp smy /can't get enough  
isp t2  
jmp .-5

/create core fields

sm4,  
    lac cmr  
    add t3 /core number  
    dac t1  
    ior (c7  
    dac t2  
    sub t3  
    dac t0  
    jdp dtc /remove attachment, if any.  
    law 0  
    lxr (c7  
    lio i uc  
    sni  
    jmp onc  
    law 1  
    lio i uc+1  
    sni  
    jmp onc  
cr0,  
    jsp gaf /user field on drum  
    jmp . /can't allocate field?  
    lac (lac  
    aam  
    dac afp  
    lac afn /field no.  
on2,  
    lor t1  
    lxr t2  
    dac i 1  
  
    idx t3 /done  
    sas t5  
    jmp sm4  
    idx smb  
    jsp lxm  
    jmp i smb  
  
smy,  
    lia  
    add (c7 uc  
    TAX  
    lac t1  
    dac i 0  
    lai  
    lxr t0  
    lio i 0  
    lxr t3  
    xct i r1  
    rcr 3s  
    CXX  
    xct i r2  
    lxr t0  
    dio i 0  
    lac (400000  
    jmp on2

/detach, core in t4, must not be real core  
/comp in t0(7), t2 = t0 + t4, t1 = t2 + 7777

/EXECUTIVE MODE

dtc, 0  
lxr t2  
lac i 1  
sza i  
jmp i dtc /empty  
dap cc4  
dzm i 1  
ior (c7  
TAX  
law 7777  
and i 1  
sas t1  
jmp .-5  
law .  
dap i 1  
aam  
lio t0  
lxr t4  
xct i r1  
law 6  
rcr 3s  
CXX  
xct i r2  
aam  
dio t0  
jmp i dtc  
cc4,

/remove attachments from core+comp in t  
/EXECUTIVE MODE

ctl,  
    0  
    lac t  
    dac t0  
    and (7770  
    TAAx  
    ior (c7  
    dac t2  
    aam  
    lio t2  
    lac t0  
    X\$AX  
    xct i r1  
    law 6  
    rcr 3s  
    CXX  
    xct i r2  
    aam  
    dio t2  
    lac t0  
    ior (c7  
    TAX  
    law 7777  
    and i 1  
    dzm i 1  
    sas t  
    jmp ct1+2  
    jmp i ct1  
  
ct2,

/create a process for computation in cmq, skip if win.

```
crp,          0
        jsp exm
        law 7777
        and cmq
        ior (c7
        TAX
        lac i con
        sma
        jmp cpn /not stopped
        dac t4 /save PRL bit
        lio i prh
        TII_<
        jmp .+10           /hoard is not empty
        aam
        lio (c7 frp       /check pool
        sni
        jmp cpn /too bad
        law i 1
        adm i prh         /increase debt
        lxr (c7 frp-prh
        lac i prh
        ior (c7
        dac t5 /7pointer to proc
        aam
        lac t5
        dac i prh         /unlink it
        lac (c7
        A↓IX /7pointer to proc
        law 7777
        and cmq
        dec i 5
        dac i prn
        ior (c7
        lia
        X→IX
        lac i prn+1
        swp
        dap i prn+1
        TAX
        lac t4
        raq 3s
        and (010000
        dac i 4 /set up PRL bit
        lai
        and (7777
        dac i prn+1
        ior (c7
        X→AX
        dap i prn
        idx crp
        jsp lxm
        jmp i crp
cpn,
```

/delete process whose number is in t4 from computation in cmr()  
/skip if win

d1p, 0  
jsp exm  
lio t4  
cmi  
lac cmr  
ior (c7  
TAX  
lac i con  
sma  
jmp dlz /not stopped  
dl7, law 7777  
and i prn  
sad cmr  
jmp dlz  
ior (c7  
X->AX  
SIIP  
inp dl7  
TXXI  
dzm i prn+1  
lxr i prn /remove from ring  
X->AX  
dap i prn  
and (7777  
ior (c7  
X->AX  
dap i prn+1  
TIIIX  
lac i prq  
ral 1s  
spa  
jmp dlw /in enter wait  
dl6, aam /delete it, 7pointer in XR  
lio (c7 frp  
dio i 0  
law 7777  
A->XA  
aam  
dac (c7 frp  
lac i prn  
dzm i prn  
and (740000  
sas (740000  
jmp dly  
law 7777 /restart for fork,  
and i dl1 /set extra process  
ior (c7  
TAX  
jmp dl6 /and return it  
dly, idx dlp  
dls, jsp lxm  
jmp i dlp  
dlw, lxr (c7 opp-prq /need to search proc chain  
dl2, law 7777  
and i prq

sad (cpp-prq  
jmp dl4  
ior (c7  
X→AX  
X\$IP  
jmp dl2 /no  
lxr i prq /yes, remove it  
X→AX  
end (7777  
dap i prq  
sas (cpp-prq  
jmp .+5  
law 7777  
A←XA  
dām  
dac (c7 cpp+1  
TIX  
dzm i prq  
jmp dl6

acr,                    lac cmr /arg processor  
                  ior (c7  
                  TAX  
                  law 7777  
                  and i con  
                  ior (c7  
                  dac cns  
                  lac ac  
                  dac t3  
                  lxr cns  
                  lia  
                  spa  
                  cma  
                  dac t2 /| AC|  
                  lac i msk  
                  dac t  
                  cma  
                  dac t1  
                  law cod  
                  dap arl  
arl,                    lac .  
                  sza i  
                  jmp ppq  
                  sad t2  
                  jmp arf  
                  law 2  
                  adm arl  
                  jmp arl  
arf,                    idx arl  
                  xct arl  
                  spi i  
                  raq 9s  
                  and (777  
                  add arn  
                  dap . 1  
                  jnp .  
  
ppq,                    law flexo q  
                  xor t2  
                  and (777770  
                  sza  
arn,                    jmp err  
                  law 7 /q1 to c7  
                  and t2  
                  sza i  
                  jmp err  
                  sub (10  
                  TAX  
                  lac (400000  
                  ral 1s  
                  SXXP  
                  jmp .-2  
                  lxr cns  
                  spi i  
                  jmp .+4  
                  cma /dismiss  
                  and i aw1  
                  jmp ppr  
                  lia /assign

```
and i aw1
A<IP
jmp yes /already assigned
lai
and xaw
sza
jmp no /can't
lah
ior i aw1
lia
xor i aw1
xor xaw
dac xaw
dio i aw1
lio t3
spi
jmp no
jmp yes

define f x,p,m
    flexo x
[p-err]*1000 m-err
terminate
```

/erq dispatch table

```
cod,      f r,pr,mr  
          f p,pp,mp  
          f af,paf,maf  
          f 1f,p1f,m1f  
          f xf,pxf,err  
          f tf,ptf,err  
          f f,pff,mff  
          f ax,pax,err  
          f sx,sax,err  
          f x,err,max  
          f q,err,mq  
          f nf,pnf,err  
          0  
  
ye3,      lac pfn /return PF in AC and skip  
          rar 6s  
          dac ac  
yes,      lac t0  
          cma  
          jmp no2  
  
no1,      dac ac  
no,       lac t0  
no2,      sma  
          jmp rtm  
          jmp rts
```

/arq routines

err,           jmp ill /arq is illegal

/+r  
pr,  
    lac t1  
    and rw  
    sza  
    jmp no  
    lac t  
    dac rw  
    lac (skp  
    dac i rr0  
    aam  
    lac (c7 rs1  
    ior (c7  
    TAX  
    aam  
    lac (c7 sr0  
    dap i 0  
    jmp yes

/-r  
mr,  
    aam  
    lac (c7 lli  
    dac i rr0  
    lac t1  
    and rw  
    dac rw  
    sza  
    jmp no  
    aan  
    lac (c7 srr  
    ior (c7  
    TAX  
    dap i 0  
    aam  
    lac (c7 rs1  
    ior (c7  
    TAX  
    aam  
    lac (c7 sr0  
    dap i 0  
    jmp no

rw,           0           / reader word

/+p  
pp,  
    lac t1  
    and pw  
    sza  
    jmp no  
    lac t  
    dac pw  
    lac (skp  
    dac i pp0  
    jmp yes

/-p  
mp,  
    aam  
    lac (c7 lli  
    dac i pp0  
    lac t1  
    and pw  
    dac pw  
    jmp no

pw,           0       /punch word

/+nf  
mf,  
pn1,  
    lxr (c7 sd  
    dzm t2  
    lac i 0  
    and t  
    sza  
    idx t2  
    SXXA  
    sas (c7 sd+26  
    jmp pn1  
    lac t2  
    rar 6s  
    jmp no1

/+tf  
ptf,  
lxr prc  
dzm i 0  
lax 77  
and io  
lxr cns  
jda pfn  
aam  
lac pfp  
and (77  
rar 6s  
jmp no1

/+xf  
pxf,  
cla  
lio io  
rcl 6s  
jda pfn  
lio pfp  
dio t  
law 77  
and io  
jda pfn  
nam  
lac i t  
lio i pfp  
dio i t  
dac i pfp  
iam  
jmp yes

/+1f  
p1f,  
    lsp gef  
    jmp no  
    lsp spf  
    jmp no  
    law 0440  
    ior afn  
    ral 6s  
    lia  
    jmp as3

/-1f  
m1f,  
    law i nuf  
    dac t2  
    lac i df1  
    ior (c7  
    dac pfp  
m1f+5,  
    law i 1  
    adm pfp  
    aam  
    lac pfp  
    sza  
    jmp m12  
    lsp t2  
    jmp m1f+5  
    jnp no  
m12,  
    and (077777  
    aam  
    sas pfp  
    jmp m12-3  
    and (77  
    rar 6s  
    jda afn  
    jmp dsf

/not a drum field

/+af  
paf,  
    ZAP

/-af  
maf,  
    law 600  
    dap dss  
    law 77  
    and io  
    sza i  
    jmp af2  
    jda pfn  
af1,  
    lac pfn  
    dac t4  
    law i 7777  
    and io  
    sza i  
    jmp af4  
af3-1,  
af3,  
dss,  
    jda afn  
    lac t4  
    jda pfn  
    skp  
    jmp asf  
    jmp dsf

af4,  
    law 77 /fill AG  
    and io  
    jda pfn  
    lac (077777  
    aam  
    and pfp  
    aam  
    sas pfp  
    jmp no /is not a field  
    sza  
    jmp af7 /PF was assigned, use its AF  
    xct dss  
    jsp gaf /PF not assigned, get new AF  
    jmp no  
    jmp af3

af7,  
    and (77  
    rar 6s  
    jmp af3-1

af2,  
    law i 7777 /fill PF  
    and io  
    sza i  
    jmp yes /both are zero  
    jda afn  
    aam  
    lac afp  
    and t  
    sza i  
    jmp af5  
    law i nuf /AF was assigned, find its PF  
    add i df1  
    ior (c7  
    d+c pfp  
    aam  
    lac pfp  
    rar 6s  
    xor (0440

sad afn  
jmp af6  
idx pfp  
and (7777  
sas i df1  
jmp apr  
jmp . /can't find it

af5, xct dss /AF was not assigned, get new PF  
jsp gpf  
jmp no  
jmp af1  
af6, jsp pfp+1  
jmp af1

/-q  
四,  
jsp dxe  
jmp no

/-f  
mff,  
jsp daf  
jmp yes

/+f  
pff,  
law i 7777  
and lo  
sza i  
jmp yes  
ral 6s  
cma  
dac t1  
dac t2  
dac t3  
jsp gaf  
jmp no  
law i 1  
adm apx  
isp t1  
jmp ga7  
jsp gp7  
jmp no  
law i 1  
adm ppx  
isp t3  
jmp gp7  
jsp gaf  
jmp .  
jsp gp7  
jmp .  
law i 77  
nam  
and i afp  
ior t  
dac i afp  
law 0440  
xor afn  
ral 6s  
dip i pfp  
iam  
isp t2  
jmp ffl  
jmp yes

ffl,

```
/+ax  
pax,          ZAP  
  
/+sx  
sax,          lac (add  
                dac t2  
                lac t1  
                and xrw  
                lla  
                and t2  
                swp  
                and (177  
                sza↓sni  
                jmp no  
                law 177  
                and xrw  
                lor t  
                lor t2  
                dac xrw  
                law 200  
                lxr cns  
                lor i aw1  
                dac i aw1  
                jmp yes  
  
/-x  
max,          lac t1  
              and xrw  
              dac xrw  
              law i 200  
              lxr cns  
              end i aw1  
              dac i aw1  
              jmp no  
  
xrw,          0          /ext. reg. word  
  
dxe,          dap dxx /dismiss external eq.  
              lxr cns  
              lac i aw1  
              xor xaw  
              dac xaw  
              law i 177  
              and i aw1  
              dac i aw1  
              jmp .  
  
dxx,  
xaw,          0          /external assignment word
```

daf, dap dfx /dismiss all fields  
law i nuf  
dac t2  
add i df1  
ior (c7  
dac pfp  
nam  
lac (700000  
and i pfp  
sza i /don't delete other capabilities  
dzm i pfp  
idx pfp  
isp t2  
jmp .-6  
lac (c7 sd  
dac afp  
lac t1  
and i afp  
dac i afp  
idx afp  
sas (c7 sd+26  
jmp .-5  
iam  
jmp .

dfx,

dsf, aam /dismiss field  
lac pfp  
rar 6s  
xor (0440  
sas afn  
jmp no  
lac t1  
nam  
and i afp  
dac i afp  
dzm i pfp  
iam  
jmp yes

ASF,

law 0440  
ior afn  
ral 6s  
lia  
lac t1  
aam  
and afp  
and (lac 177  
sza  
jmp no /can't get it  
aam  
lac pfp  
sza  
jmp at2  
lac t  
aam  
and afp  
sza  
jmp no  
law 177

as3,

nam  
and i afp  
ior t  
dac i afp  
dio i pfp  
iam  
jmp yes

as2, A\$IP|  
jmp as3  
jmp no

afp, 0 /afp>afn  
dap apx  
afp+2, lac (-c7-sd+1  
add afp  
rar 6s  
dip afn  
jmp .  
  
afn, 0 /afn>afp  
dap anx  
lac afn  
sza i  
jmp no  
ral 6s  
sub (27  
sma  
jmp no  
add (c7 sd 26  
d+p afp  
jmp .  
  
anx, 0 /pfp>pfn  
dap ppx  
pfp+2, lac pfp  
sub i df1  
add (nuf-c7  
dac pfn  
jmp .  
  
pfn, 0 /pfn>pfp  
dap pnx  
lac pfn  
sza i  
jmp no  
sub (nuf  
sma  
jmp no  
add i df1  
ior (c7  
dac pfp  
jmp .  
  
px, 0 /get absolute field  
dap apx /get absolute field  
lac (c7 sd  
dac afp  
gaf+3, aam  
lac afp  
sza i  
jmp ga2  
ga7, idx afp  
sas (c7 sd 26  
jmp gaf 3  
jmp apx  
ga2, idx apx  
jmp afp 2  
  
gpf, 0 /get pseudo field  
dap opx /get pseudo field  
law i nuf-1  
add i df1

```
for (c7  
dac pfp  
aam  
lac pfp  
sza i  
jmp gp2  
idx pfp  
and (7777  
sas i df1  
jmp gpf 5  
jmp ppx  
idx ppx  
jmp pfp 2  
gp2,  
gp7,  
gpf+5,
```

```
/create capability

cra,          0
    dap cax
    law 7777
    and cmp /crock so create sphere will work
    jdp fcl
    sza i
    jmp cax /no C-list
    sub (1
    and ac
    dac t2 /requested index number
    dac t1
ca1,          aed clp
TAX
    lac i 0
TAP|
jmp ca2
sas t2 /not free
cax,          jmp .
idx t1 /asked for zero
sas cls
jmp ca1
jmp cax
ca2,          lac cra
dac i 0
lac t1
dac ac
idx cax
jmp cax
```

/meta instructions and ivks  
/transmitted word = +0 for arg, -0 for arg 1,  
/20\_<T\_<77 for meta, T\_>100 for ivk w/c7 C-list  
/T<0 for c0 C-list (aedr-70000)

atm,            iam  
          eem  
          TAX  
          law i 2  
          sas ff  
          jmp , -1  
          isp ff  
          jmp , +4  
          law i 1  
          adm ff  
          jmp , -7  
          dio t0  
          TXA  
          dap enp /set up entered process  
          ior (10000  
          TAX  
          law 7777  
          and i 0  
          ior (c7  
          dac prc  
          TAX  
          sad i prn+1  
          jmp ill /process has been abandoned  
          lac i 5  
          dac cmr  
          ior (c7  
          TAX  
          lio i con  
          law 21  
          spi  
          jmp enp /sphere is stopped  
          lio (ac  
          law 1  
          xct enp /read live registers  
          lac t0  
          TAA|=  
          jmp aqr  
          and (77  
          sas t0  
          jmp iv2 /ivk  
mtb,            law i 20                    /meta  
          add t0  
          TAAX>P  
          jmp ill  
          sub (30  
          sma  
          jmp ill /illegal number  
          lac cmr /computation to which meta refers  
          ior (240000  
          aam  
          dac (i 71  
          lac cmr  
          ior (c7  
          dac cmp

lac ac  
xct i .+1  
jmp m20  
jmp m21  
jmp m22  
jmp m23  
jmp m24  
jmp m25  
jmp m26  
jmp m27  
jmp ill  
jmp ill  
jmp m32  
jmp m33  
jmp ill  
jmp m35  
jmp m36  
jmp m37  
jmp m40  
jmp m41  
jmp m42  
jmp m43  
jmp m44  
jmp m45  
jmp m46  
jmp m47

ill,  
enp,  
law 41  
inv  
law i 1  
adm ff  
qit

ac,  
io,  
ff,  
0  
0  
-2

/mta 200 - read illegal instruction return

m20,            lxr cmp  
          lac i ilr  
          dac ac

rtm,            lio (ac /return, no skip  
          law 11  
          xct enp  
          law 31  
          jmp enp

/mta 201 - set illegal instruction return

m21,            lxr cmp  
          dac i ilr  
          jmp rtm

/mta 202 - read memory protection return

m22,            lxr cmp  
          lac i imr  
          jmp m20+2

/mta 203 - set memory protection return

m23,            lxr cmp  
          dac i imr  
          jmp rtm

/mta 204 - release capability

m24,  
    lac cmr  
    jdp fcl  
    sza i  
    jmp rtm /no C-list  
    sub (1  
    and ac  
    add clp  
    TAX  
    lac i 0  
    and (700000  
    sza i  
    jmp rtm  
    lac i 0  
    dzm i 0  
    jda rmv  
    jmp rtm

/mta 205 - detach

m25,  
    and (070000  
    ral 6s  
    sza i  
    lac ac  
    and (7  
    dac t4  
    sub (6  
    sma  
    jnp rts /if illegal, ok  
    lac cmr  
    add t4  
    dac t1  
    ior (c7  
    dac t2  
    sub t4  
    dac t0  
    lxr t2  
    law i 7777  
    and i 1  
    sza  
    jmp rtm /is a real core, fail  
    jsp exm  
    jdp dtc  
    jsp lxm  
    jmp rts

/mta 206 - read memory bound

m26,  
    law i 6  
    dac t3  
    dzm t1  
    dzm t2  
    lxr cmp  
rd1,  
    lac t2  
    adm t2  
    law i 7777  
    and i 1  
    sza i

```
jnp rd4
idx t1 /real core
SXX
lsp t3
jmp rd1
lac t2 /attachments
ral 6s
ior t1 /memory bound
rar 6s
dac ac
jmp rtm
rd4,
lac i 1
sza
idx t2
jmp rd3
```

/mta 207 - set memory bound

```
m27,      and (070000
ral 6s
sza i
lac ac
and (7
dac t5
sza
sub (7
sma
jmp rtm
jdp smb
jmp rtm
jmp rts
```

/mta 302 - create sphere

```
m32,      law 7777
and cmp
dac cmq
lac frc
sza i
jmp rtm /can't
mul (30>2
lai
add (cms-ccf*30
dac cmr
ior (c7
TAX
z=1
repeat 6,dzm i z    z=z+1
lio (666666
dio i 0
law 1
dac t5 /initial memory bound
jdp smb
jmp rtm
lac cmr
ior (c7
TAX
lac (400000
dac i bp1
dzm i bp2
dzm i bp3
lac cmq
dac i sup
lac io
and (077777
dac i spe
clc
dac i ilr
dac i imr
dzm i bel
lac (400000
dac i con
dzm i prh
dzm i quo
dzm i quo+1
lac cmr
dac i prn
dac i prn+1
ior (240000
jda cra
jmp ,+7
lxr frc
lio i 0
law 1
dac i 0
dio frc
jmp rts
dzm t5
jdp smb
nop
jmp rtm
```

/mta 303 - create queue

m33,  
    lac frq  
    sza i  
    jmp rtm /can't set one  
    sal 1s  
    add (-prq+340000-qqf\*2  
    aam  
    add (c7 qqf  
    jda cra  
    jnp rtm /can't find space  
    lax 7777  
    and cra  
    TAAAX  
    ior (c7  
    X>AX  
    dac i prq+1                 /initialize it  
    lio lo  
    TII<  
    CII=  
    lai  
    dac i prq  
    lxr frq  
    lio i 0  
    law 1  
    dac i 0  
    dio frq  
  
rts,  
    lio (ac /return, skip  
    law 11  
    xct enp  
    law 51  
    jmp enp

/mta 305 - create file

m35,            lac frf  
          sea i  
          jmp rtm  
          mul (3  
          scr ls  
          lai  
          add (ffs+540000-ffff\*3  
          jda cra  
          jmp rtm  
          lxr frf  
          lio i 0  
          law 1  
          dac i 0  
          dio frf  
          jmp rts

/mta 306 - create IO device

m36,            lac ac  
          ral 6s  
          and (7  
          TAX  
          lac ac  
          rar 6s  
          TXP|  
          jmp 36a  
          jmp 36b  
  
36a,            and (3  
          dac t4  
          TAAK  
          lio i mtu  
          sni i  
          jmp rtm /not available  
          ral 3s  
          ior (714000  
          jda cra  
          jmp rtm  
          lxr t4  
          idx i mtu  
          jmp rts  
  
36b,            and (77  
          ior (600000  
          jda cra  
          jmp rtm  
          jmp rts  
  
m37,            jmp III

/mta 400 - read capability

m40,            lac cmr  
          jdp fcl  
          sza i  
          jmp rtm /no C-list  
          sub (1  
          and ac  
          add clp  
          TAX  
          lac i 0  
          dac ac  
          jmp rts

/mta 401 - exchange capabilities

m41,            lac cmr  
          jdp fcl  
          sza i  
          jmp rtm /no C-list  
          sub (1  
          lia  
          and ac  
          add clp  
          dac t1  
          lac ac  
          rar 6s  
          A<-IA  
          add clp  
          TAX  
          lac i 0  
          aam  
          llo t1  
          aam  
          dac t1  
          dio i 0  
          jmp rts

/mta 402 - turn PRL off

m42, lxr cmp  
lac (200000  
adm i con  
dac t7  
rat 2s  
sma  
jmp 42d /already off  
law 2  
ivk 71 /suppress processing  
lac cmr  
ior (240000  
lem  
jda gcp  
dac t8  
dzm t8+1  
stf 5  
lem  
lxr (t8  
jdp ssr  
jmp .+4  
lac gcp  
jda rmv /doesn't mind lem  
jmp .-7 /returns in eem  
eem

/cmr is now clobbered

42b, lxr cmp  
lac (-100000  
adm i con  
law 7777  
and i prn  
ior (c7  
sad cmp  
jmp 42c  
TAX  
lac (-010000  
end i 4  
dac i 4  
jmp 42b  
42c, lio t7  
law 12  
spi i  
ivk 71  
42d, lxr cmp  
lac (-200000  
adm i con  
jmp rtm

/mta 403 - turn on PRL

m43,  
    lac cmr  
    ior (c7  
    TAX  
    lac (200000  
    adm i con  
    dac t3  
    ral 2s  
    spa  
    jmp 43d /already on  
    law 2  
    ivk 71 /suppress processing  
    jsp exm  
    lac cmr  
    ior (c7  
    TAX  
    lac (100000  
    adm i con  
43b,  
    law 7777  
    and i prn  
    sad cmr  
    jmp 43c  
    ior (c7  
    TAX  
    lac (010000  
    ior i 4  
    dac i 4  
    jmp 43b  
    law 43f  
    dac ct1  
    lac cmr  
    dac t  
    ior (c7  
    TAX  
    law 7777  
    and i 1  
    X→AH  
    dap i 1  
    lai  
    jmp ct2 /force off attachments (too bad)  
43f,  
    jtp lxm  
    law 22  
    lio (020000  
    ivk 71 /attach  
    die  
    TIX  
    dm i 0 /clear C-list  
    SXXA  
    sas (020100  
    jmp .-3  
    lio t3  
    law 12  
    spi i  
    ivk 71 /permit processing  
43d,  
    lac cmr  
    ior (c7  
    TAX  
    lac (-200000

adm i con  
jmp rtm

m44,  
m45,  
jmp ill  
lxr cmp /405 - read hoard size  
lac i prh  
dac ac  
spq  
jnp rtm  
dzm ac  
sza i  
jmp rtm  
ior (c7  
TAX  
idx ac  
lac i 0  
jmp .-6  
m46,  
lxr cmp /406 - decrease hoard  
lac i prh  
szm  
jmp .+4  
law i 1  
adm i prh  
jmp rtm  
dac t0  
ior (c7  
dac t1  
aam  
lio t1  
dio i prh  
aam  
lac (c7 frp  
aam  
dac t1  
lac t0  
aam  
dac (c7 frp  
jmp rtm  
m47,  
lxr cmp /407 - increase hoard  
lac i prh  
sma  
jnp .+3  
idx i prh  
jmp rts  
aam  
lio (c7 frp  
sni  
jmp rtm /too bad  
dio i prh  
swp  
ior (c7  
TAX  
lac i 0  
dio i 0  
aam  
dac (c7 frp  
jnp rts

/translate capability word in AC into object pointer  
/in AC, tt0

gcp, 0  
dap gpx  
cli  
lac gcp  
rcl 3s  
law 7777  
and gcp  
TIX  
xct i ,+1  
die  
jmp gs1  
jmp gs2  
jmp gs3  
die  
jmp gs5  
jmp gs6  
jmp gs7

gs1, sub (cms-[ppf+100]\*13 /entered process  
mul (1  
div (13  
die  
sni i  
die  
TAAK  
lio gcp  
ril 5s  
law 1  
/spi  
dac i 0 /set ref, count to 1  
TXA  
ior (020000  
jmp ssy

gs2, sub (cms-ccf\*30 /sphere  
mul (1  
div (30  
die  
sni i  
die  
ior (040000  
jmp ssy

gs3, sub (-prq-qaf\*2 /queue  
lxr (c7  
sub i qqt  
mul (1  
div (2  
die  
sni i  
die  
ior (060000  
jmp ssy

gs5, sub (ffs-fff\*3 /file  
mul (1  
div (3  
die  
sni i  
die  
ior (120000  
jmp ssy

gs6, and (77 /hardware device  
add (lif  
TAAX  
lio (1  
dio i 0 /make reference count = 1  
ior (140000  
jmp ssy

gs7, rar 3s /magnetic tape  
and (3  
add (mtu  
ior (160000  
dac ;tt0  
law 377  
aam  
and tt0  
sza i  
die /count=0  
lac tt0  
jnp .

ssy,

gpx,

/get sub-object, <XR> has object pointer, <XR+1> has  
/index of sub-object, initially 0, gets advanced to  
/next one, skips if sub-object available, pointer in AC  
/F5 → remove it  
/XR saved

ssr, 0  
TXXA  
dac ;xrs  
lac i 0  
ral 2s  
cli  
rcl 3s  
TIX /type  
sar 5s /pointer  
xct i .+1  
  
die  
jmp ssx /1 - entered process  
jmp sb2 /2 - sphere  
jmp ssx /3 - queue  
die  
jmp ssx /5 - file  
jmp ssx /6 - hardware device  
jmp ssx /7 - entry  
  
sb2, mul (15  
lai  
add (cms-ccf\*30  
eem  
jdp fcl  
lem  
sbb, lxr xrs  
lac i 1  
sub cls  
sma  
jmp ssx  
idx i 1  
sub (1  
add clp  
TAX  
lac i 0  
and (700000 /to ignore drum fields  
sza i  
jmp sbb  
lac i 0  
dac t1  
szf 5  
dam i 0  
jda gcp  
idx ssr  
lac tt0  
lxr xrs  
jmp i ssr

/remove reference to capability given in AC  
/must have already been removed from C-list

rmv, 0  
dap rmx  
lem  
lac rmv  
jda gcp  
d+c ;tt1  
law pdl /clear pdl completely  
dac cpd

/begin mark phase

gm0, lxr cpd /clear pdl down to thing being deleted  
clf 6  
clf 5  
law 377  
aam  
and tt1  
sad (1  
jmp gvn  
sza i  
die /count=0  
lac tt1  
dac i 0  
law 400  
aam  
adm i 0 /mark it  
and (177400  
sza i  
die /mark overflow  
sas (400  
jmp gm2  
dem i 1 /first time  
lac (177777  
aam  
and i 0  
aam  
dac i 0  
gm3, jep ssr  
jmp gm2  
dac i 2  
law 2  
A+XXA  
sad (pde  
die /pdl overflow  
jmp gm1  
lac cpd  
X\$AP|  
jmp gg1 /empty, begin check phase  
law i 2  
X+AX  
jmp gm3

/check for external references to A

gg1,            aam       /check B  
          lac i 0  
          spa  
          jmp gg9 /P[B] already on, return  
          and (177777  
          cli  
          rcr 8s  
          ril 8s /mark in AC, count in IO  
          sza i  
          jmp gg4 /checked already, continue only if turning on P[B]  
          AMIA\_<  
          die      /mark > count  
          and (400000       /sign bit if external reference  
          ior i 0  
          dac i 0 /turn on F  
          sma  
          jmp gg3  
gg2,            and (177777       /B has external reference  
          sad tt1  
          stf 6 /B = A, can't erase A  
          lac (400000  
          aam  
          ior i 0  
          eam  
          dac i 0 /I → P[B]  
gg3,            dzm i 1 /sub-object index  
          lac (-177400  
          aam  
          and i 0  
          aam  
          dac i 0 /clear mark field  
gg5,            jdp ssr /get sub-object  
          jmp gg8 /ran out  
          lia  
          lac i 0  
          and (400000  
          A+II      /preserve F during push  
          law 2  
          A+XXA  
          sad (pde  
          die  
          dio i 0  
          jmp gg1 /recurse  
gg4,            lac i 0  
          spa  
          jmp gg2  
gg9,            lac cpd /pop, restore F  
          X\$AP|  
          jmp gg6 /finished  
          law i 2  
          A+XX  
          jmp gg5

g56,  
lac i 0  
and (177777  
sas tt1  
die  
gwn,  
tax i 1  
aam  
adm tt1 /decrease count  
saf 6  
jmp sk9 /it has external references

lac tt1 /can delete it  
dac i 0  
dzm i 1  
sk0,  
stf 5  
jdp ssr /find and erate a sub-object  
jmp gh3  
dac tt1  
law 2  
adm cpd  
sae (pde  
die  
jmp gm0 /erase it

gh3,  
law 377  
aam  
and i 0  
sza  
die /count not 0  
lac i 0  
llo cpd  
lxr (pdl  
X\$IP|  
jmp .+6  
sad i 0  
jmp sk9 /is on pdl from before  
SXX  
SXX  
jmp , -6

/delete object in AC

```
dac tt0
ral 2s
cli
rcl 3s
TIX
law 7777
and tt0
mul i gzz
scr 1s
lai
add i gyy
eem
aam
dac (010076
lem
law 7777
and tt0
xct i .+1
die
jmp gk1
jmp gk2
jmp gk3
die
jmp gk5
jmp gk6
jmp gk7

gk1,
law 41 /process
ivk 76
jmp gk9

gk2,
TAXI /sphere
lac frc
dac i 0
dio frc
law 2
ivk 76 /stop
eem
aam
lac (010076
and (7777
dac cmr
dzm t5
jdp smb /set memory bound to 0
nop
dzm t4
jsp exm
lac cmr
add t4
dac t1
lcr (c7
dac t2
sub t4
dac t0
jdp dtc /remove attachment
idx t4
sas (6
```

jmp gq2  
jsplxm  
law 1  
dac t4  
jdp dlp /delete first process  
skp i  
jmp .-2  
jspxm /clean off process board  
lac cmr  
ior (c7  
TAX  
lac i prh  
spq  
jmp gq5  
dac t0  
and (7777  
ior (c7  
TAX  
lac i 0  
sza  
jmp .-5  
aam  
lac (c7 frp  
dac i 0  
lac t0  
aam  
dac (c7 frp  
gq5,  
jsplxm  
lem  
jmp gk9

gk3,  
TAXI /queue  
lac frq  
dac i 0  
dio frq  
lxr (c7  
lac i qqt  
lxr (010076  
adm i 0  
law 23  
ivk 76 /empty it  
jmp .-1  
jmp gk9

gk5,  
TAXI /file  
lac frf  
dac i 0  
dio frf  
jmp gk9

gk6,  
jmp gk9 /hardware device

gk7,  
TAXI /mag tape  
dam i 0

gk9,  
law i 2  
adm cpd  
TAXX  
sas (pd1-2  
jmp gk0

rmx ,  
eem  
jmp .

622 ,  
0  
13  
30  
2  
0  
3  
0  
10

gvy ,  
0  
cms-[ppf+100]\*13+050000  
cms-ccf\*30+240000  
-prq-qqf\*2+340000  
0  
ffs-fff\*3+540000  
0  
-mtu\*10+714000

iv2,  
lac t0  
spa  
jmp .+5  
add (c7  
TAX /absolute pointer to capability  
llo i 0  
jmp iv3  
lac cmr /core 0 C-list  
ior (240000  
aam  
dac (i 76  
law buf  
mta  
cli  
lac (010102  
ivk 76  
jmp .-1  
lac (070000  
add t0  
TAX  
llo i buf  
dic t /capability word  
law 7777  
A←IAP|  
jmp ill  
dac cmq /low 12 bits of capability  
cla  
rcl 3s  
TAAXI  
xor ac  
rcr 3s  
A→IP  
jmp ill /improper code  
xct i ,+1

jnp ill /0  
jmp ill /1  
jmp ifs /2 - sphere  
jmp ill /3  
jmp ill /4  
jmp ill /5  
jmp ill /6  
jmp ill /7

ifs,  
and (77 /sphere ivk  
TAX  
lac cmr  
dac cmp /sphere executing the ivk  
lac cmq  
dac cmr /0pointer to sphere  
ior (c7  
dac t1 /7pointer to sphere  
law 60  
A←XP|  
jmp ill  
lax 7  
A←XX  
xct i ,+1  
jmp c20 /202 - create process

jmp c21 /212 - delete process  
jmp c22 /222 - count processes  
jmp ill  
jmp c24 /242 - antigrant  
jmp c25 /252 - grant  
jmp ill  
jmp c27 /272 - execute meta

c20, jdp crp  
jmp rtm  
jmp rts

c21, lio io  
dio t4  
jdp dlp  
jmp rtm  
jmp rts

c22, jsp exm  
lac cmq  
ZIP  
SII  
ior (c7  
TAX  
law 7777  
and i prn  
ses cmq  
jmp .-6  
dio ac  
jsp lxm  
jmp rtm

c24,            lac cmr  
          jmp ,+4

c25,            lio cmr  
          lac cmp  
          dio cmp /receiver  
          lio io  
          dio ac  
          jdp fcl /sender  
          sza i  
          jmp rtm /no C-list in sender  
          sub (1  
          lio ac  
          rir 6s  
          A←IA  
          add clp  
          TAX  
          lac i 0  
          dac cra /capability to be moved  
          and (7777  
          sza i  
          jmp rtm /nothing  
          lac cra  
          and (700000  
          sza  
          sad (100000  
          jmp rtm /can't do this yet  
          jsp cra+1  
          jmp rtm /couldn't receive it  
          lem  
          lac cra  
          jda gcp  
          aam  
          idx tt0 /index count  
          eem  
          jmp rts

c27,            lac ac  
          ral 9s  
          ior (70  
          lia  
          rar 3s  
          A←IA  
          and (77  
          dac t0 /code number  
          lxr prc  
          lac i d11  
          dac ac /new AC  
          jmp mtb

/find C-list of computation in AC(0)  
/returns pointer to origin in clp  
/returns size (0,20,or 100) in cls, AC

fcl, 0  
    lcr (240000  
    aam  
    dac (010073  
    and (7777  
    lcr (c7  
    TAX  
    law 100  
    dac cls  
    lac i con  
    rat 2s  
    spa  
    jmp f13 /core 0 C-list  
    dzm cls  
    law 7777  
    and i con  
    sza i  
    jnp f14 /no C-list  
    lcr (c7 /core 7 C-list  
    TAX  
    law nuf  
    dac cls  
    lac i df1  
    add (c7-nuf  
    dac clp  
    lac cls  
    jmp i fcl  
  
f14,  
    llo (020000                    /core 0 C-list  
    law 22  
    ivk 73 /attach  
    die  
    lac (020000  
    jmp f14  
  
dimension clt(1),clp(1)

/enter executive mode

exm, dap ex9  
lxr (c7  
lok  
lio i 107  
dio ;ex1  
lio i 110  
dio ;ex2  
lio i 111  
dio ;ex3  
lio (eem  
dio i 107  
lio (jmp i 111  
dio i 110  
law ex8  
dac i 111  
hlt  
lx9, jmp .  
ex8, eem  
lxr (c7  
lio ex1  
dio i 107  
lio ex2  
dio i 110  
lio ex3  
dio i 111  
ex9, jmp .

/leave executive mode

lxm, dap lx9  
jmp i (70000+ubs

constants  
printc ,+3200 printc 77  
variables  
buf,  
start